

AMENDMENTS

Please amend the above-referenced application as follows:

IN THE CLAIMS:

Please amend the claims as indicated below.

- et
- 1 1. (Twice Amended.) An apparatus for performing single instruction
2 multiple data instructions using a single multiply accumulate (MAC) unit,
3 comprising:
4 a single multiply accumulate (MAC) unit configured to generate a data result
5 responsive to a single-instruction multiple-data (SIMD) instruction, the data result
6 having a first half and a second half;
7 a register communicatively coupled to the multiply accumulate single MAC
8 unit, the register configured to store the first half of the data result; and
9 a miscellaneous-logic unit configured to initiate the release of the first half of
10 the data result from the register to synchronize the first half of the data result with the
11 second half of the data result.
- 12 2. (Previously Amended.) The apparatus of claim 1, wherein said
13 MAC unit generates the first half of the data result before the second half of the data
14 result.
- 1 3. (Previously Amended.) The apparatus of claim 2, further
2 comprising:
3 a register file configured to receive the first half of the data result substantially
4 concurrently with the second half of the data result.
- 1 4. (Twice Amended.) The apparatus of claim 3, wherein said
2 miscellaneous-logic (MAC) unit generates an exception result when said
3 miscellaneous-logic unit determines the first half of the data result is erroneous.

CT
1 5. (Twice Amended.) The apparatus of claim 4, wherein said
2 miscellaneous-logic (MAC) unit is configured to initiate the release of one of the first
3 half of the data result stored in said register or initiate the release of the exception
4 result.

1 6. (Twice Amended.) A method for performing single-instruction
2 multiple-data (SIMD) instructions using ~~a single multiply-accumulate unit~~,
3 comprising the steps of:
4 providing a single multiply-accumulate (MAC) unit configured to generate a
5 first half of a data result and a second half of a data result responsive to a single-
6 instruction multiple-data (SIMD) instruction;
7 applying the first half of the data result at an input of a register;
8 using a miscellaneous-logic unit to generate an exception result; and
9 applying the first half of the data result and the second half of the data result at
10 an input of a buffer when the miscellaneous-logic unit determines that the first half of
11 the data result and the second half of the data result are valid, otherwise applying an
12 exception result at the input of the buffer when the miscellaneous-logic unit
13 determines that the first half of the data result and the second half of the data result are
14 invalid.

1 7. (Previously Amended.) The method of claim 6, further
2 comprising:
3 latching a first operand into said MAC unit; and
4 latching a second operand into said MAC unit.

1 8. (Previously Amended.) The method of claim 7, further
2 comprising:
3 generating the first half of the data result from the first operand; and
4 generating the second half of the data result from the second operand.

1 9. (Previously Amended.) The method of claim 8, further
2 comprising:
3 latching the first half of the data result in a miscellaneous logic unit; and
4 latching the second half of the data result in the miscellaneous logic unit.

1 10. (Previously Amended.) The method of claim 9, further
2 comprising:
3 generating the exception result from the first half of the data result and the
4 second half of the data result.

1 11. (Twice Amended.) An apparatus for performing single-instruction
2 multiple-data (SIMD) instructions, comprising:
3 a single multiply-accumulate (MAC) unit for generating a first data result
4 responsive to a first operand and a second data result responsive to a second operand,
5 wherein said first and second operands are associated with a SIMD instruction;
6 means for storing the first data result;
7 ~~means for generating a second data result responsive to a second operand;~~
8 means for generating an exception result responsive to the first and second
9 data results;
10 means for forwarding the first data result and the second data result to a buffer
11 when the exception result indicates that the first data result and the second data result
12 are valid; and
13 means for communicating the exception to the buffer when the means for
14 forwarding indicates that the first and second data results are invalid.

1 12. (Previously Amended.) The apparatus of claim 11, further
2 comprising:
3 means for storing the first operand; and
4 means for storing the second operand.